

CLAIMS

- 1 1. A method for providing an on-disk representation of a named data stream associated
2 with at least one of a directory inode and file inode of a file system of a network storage
3 appliance, the method comprising the steps of:
4 allocating a first free inode of the file system;
5 inserting a `stream_dir` type into a first type field of the first free inode to thereby
6 convert the first free inode to a stream directory inode, the stream directory inode having
7 a first data section with a first pointer configured to reference a stream directory data
8 block;
9 modifying a `xinode` field of the file inode to reference the stream directory inode;
10 allocating an entry of the stream directory data block to store a name of the named
11 data stream, the allocated entry further allocating a second free inode of the file system;
12 and
13 inserting a stream type into a second type field of the second free inode to convert
14 the second free inode to a stream inode for the named data stream, the stream inode hav-
15 ing a second data section with a second pointer that references a user data block associ-
16 ated with the named data stream.
- 1 2. The method of Claim 1 further comprising the step of assigning a third pointer to the
2 allocated entry of the stream directory data block that references the stream inode.
- 1 3. The method of Claim 1 wherein the step of allocating a first free inode comprises the
2 step of asserting a predetermined flag within the file inode to transform the file inode into
3 a base inode having more than one data fork.
- 1 4. The method of Claim 3 wherein the predetermined flag is a `WAFL_flag_stream` flag.
- 1 5. The method of Claim 1 further comprising the steps of:

2 loading a root inode and its referenced data blocks from a disk into a memory of
3 the network storage appliance; and
4 searching contents of the root inode data blocks for a directory name of the di-
5 rectory inode.

1 6. The method of Claim 5 further comprising the steps of:
2 upon finding the directory name, loading the directory inode and its referenced
3 data blocks into memory; and
4 searching contents of the directory inode data blocks for a filename of the file
5 inode.

1 7. The method of Claim 6 further comprising the step of, upon finding the filename,
2 loading the file inode and its referenced data blocks into the memory.

1 8. A system for providing on-disk representations of multiple named data streams within
2 a filer, the system comprising:
3 a processor;
4 a memory coupled to the processor and having locations addressable by the proc-
5 essor;
6 at least one disk coupled to the memory and processor; and
7 an operating system resident in the memory locations and invoking storage op-
8 erations in support of a file system configured to logically organize information as a hier-
9 archical structure of directory and file inodes on the disk, each named data stream stored
10 on the disk as a representation embodying a stream inode associated with a file inode,
11 each on-disk file inode having a default data stream and at least one named data stream
12 inode.

1 9. The system of Claim 8 further comprising a storage adapter interconnected with the
2 processor, memory and disk, the storage adapter cooperating with the operating system to
3 access the information stored on the disk.

1 10. The system of Claim 8 further comprising a network adapter coupled to the processor
2 and memory of the filer, the network adapter connecting the filer to a client over a com-
3 puter network, the client interacting with the filer by exchanging packets encapsulating a
4 record requesting file services from the filer using a file system protocol over the net-
5 work.

1 11. The system of Claim 10 wherein the file system protocol is a Common Internet File
2 System (CIFS) protocol and wherein the record is a CIFS record comprising information
3 pertaining to an operation directed to the named data stream.

1 12. The system of Claim 11 wherein the operating system comprises a series of software
2 layers, including a file system protocol layer configured to support the CIFS protocol and
3 a Write Anywhere File Layout (WAFL) layer configured to implement the file system.

1 13. The system of Claim 12 wherein the CIFS record is interpreted as directed to a
2 named data stream associated with a file and transformed into a message structure by the
3 CIFS layer, and further passed to the WAFL layer, where the operation is performed.

1 14. The system of Claim 13 wherein the message is passed from the CIFS layer to the
2 WAFL layer as a function call.

1 15. The system of Claim 13 wherein the WAFL layer loads the stream inode from disk
2 into memory and accesses the stream inode as instructed by the operation.

1 16. The system of Claim 12 wherein the operating system further comprises a media ac-
2 cess layer of network drivers, network protocol layers, a disk storage layer that imple-
3 ments a disk storage protocol and a disk driver layer that implements a disk access proto-
4 col.

1 17. The system of Claim 16 wherein a storage access request data path through the oper-
2 ating system layers enables performance of data storage access for the client request re-
3 ceived at the filer.

1 18. The system of Claim 17 wherein the storage access request data path is implemented
2 as logic circuitry embodied within a hardware circuit.

1 19. A computer readable medium containing executable program instructions for pro-
2 viding an on-disk representation of a named data stream associated with at least one of a
3 directory inode and file inode of a file system, the executable program instructions com-
4 prising program instructions for:

5 allocating a first free inode of the file system;
6 inserting a stream_dir type into a first type field of the first free inode to thereby
7 convert the first free inode to a stream directory inode, the stream directory inode having
8 a first data section with a first pointer configured to reference a stream directory data
9 block;
10 modifying a xinode field of the file inode to reference the stream directory inode;
11 allocating an entry of the stream directory data block to store a name of the named
12 data stream, the allocated entry further allocating a second free inode of the file system;
13 and
14 inserting a stream type into a second type field of the second free inode to convert
15 the second free inode to a stream inode for the named data stream, the stream inode hav-
16 ing a second data section with a second pointer that references a user data block associ-
17 ated with the named data stream.

1 20. Apparatus for providing an on-disk representation of a named data stream associated
2 with at least one of a directory inode and file inode of a file system of a filer, the appara-
3 tus comprising:

4 means for allocating a first free inode of the file system;

5 means for inserting a stream_dir type into a first type field of the first free inode
6 to thereby convert the first free inode to a stream directory inode, the stream directory
7 inode having a first data section with a first pointer configured to reference a stream di-
8 rectory data block;
9 means for modifying a xinode field of the file inode to reference the stream di-
10 rectory inode;
11 means for allocating an entry of the stream directory data block to store a name of
12 the named data stream, the allocated entry further allocating a second free inode of the
13 file system; and
14 means for inserting a stream type into a second type field of the second free inode
15 to convert the second free inode to a stream inode for the named data stream, the stream
16 inode having a second data section with a second pointer that references a user data block
17 associated with the named data stream.

1 21. A storage system for providing on-disk representations of multiple named data
2 streams within a computer, the system comprising:

3 an operating system resident in a memory and invoking storage operations in sup-
4 port of a file system configured to logically organize information as a hierarchical struc-
5 ture of directory and file inodes on the disk, each named data stream stored on the disk as
6 a representation embodying a stream inode associated with a file inode, each on-disk file
7 inode having a default data stream and at least one named data stream inode.

1 22. A method for accessing an on-disk representation of a named data stream associated
2 with at least one of a directory inode and file inode of a file system of a server computer,
3 the method comprising the steps of:

4 sending a packet from a client to the server over a network medium, the packet
5 encapsulating a record requesting a file service directed to the named data stream;

6 creating a message from the record and passing the message to a file system layer
7 of the server;

8 loading a stream inode and a referenced data block for the named data stream
9 from a disk into a memory of the server; and
10 processing the stream inode in accordance with the requested file service.

1 23. The method of Claim 22 wherein the requested file service is a write operation.

1 24. The method of Claim 23 further comprising the steps of:

2 loading a base inode of the stream inode from the disk into the memory;
3 updating a modification time stamp stored in the base inode for the named data
4 stream;
5 updating the referenced data block as instructed by the write operation;
6 marking the referenced data block, the base inode and the stream inode dirty; and
7 flushing the marked block and inodes to the disk.

1 25. The method of Claim 22 wherein the requested file service is a read operation.

1 26. The method of Claim 25 further comprising the steps of:

2 loading a base inode of the stream inode from the disk into the memory;
3 updating an access time stamp stored in the base inode for the named data stream;
4 retrieving data contained in the referenced data block for delivery to the client;
5 marking the base inode dirty; and
6 flushing the marked inode to the disk.

1 27. A method for accessing an on-disk representation of a named data stream associated
2 with at least one of a directory inode and file inode of a file system of a server computer,
3 the method comprising the steps of:

4 sending a packet from a client to the server over a network medium, the packet
5 encapsulating a record requesting a file service directed to the named data stream;

6 creating a message from the record and passing the message to a file system layer
7 of the server;
8 loading a base inode referenced by a file handle of the message from a disk into a
9 memory of the server;
10 determining whether a predetermined flag of the base inode is asserted and, if so,
11 loading a stream directory inode referenced by the base inode, along with a stream di-
12 rectory block referenced by the stream directory inode, into the memory;
13 loading a stream inode for the named data stream referenced by the stream direc-
14 tory block into the memory; and
15 retrieving (i) a name of the stream inode from the stream directory block and (ii) a
16 size of the stream inode from the stream inode for delivery to the client.

1 28. A multi-protocol data access storage system for providing on-disk representations of
2 multiple named data streams within a computer, the system comprising:

3 an operating system resident in a memory of the computer and invoking storage
4 operations in support of a file system configured to logically organize information as a
5 hierarchical structure of directory and file inodes on the disk, the operating system in-
6 cluding a file system protocol layer configured to provide data access in support of a plu-
7 rality of file system protocols, each named data stream stored on the disk as a representa-
8 tion embodying a stream inode associated with a file inode, each on-disk file inode hav-
9 ing a default data stream and at least one named data stream inode.

1 29. The multi-protocol data access storage system of Claim 28 wherein one of the file
2 system protocols is a Common Internet File System (CIFS) protocol.